Comparison Analysis of Post- Processing Method for Punjabi Font

Dharam Veer Sharma1 Sandeep Kaur2
1 punjabi university Patiala district Patiala
2 house no 70,v.p.o Ghanaur ,district Patiala
dveer72@hotmail.com1, gurayasandeep@gmail.com2

Department of Computer Science, Punjabi University, Patiala, INDIA

Abstract: *Offline handwritten character recognition involves generation of text from OCR (optical character recognition) systems. Due to distortion and noise present in documents, it becomes difficult to recognize words. Offline handwritten character recognition involves the development of such a system different people has different writing style. This paper therefore proposed the correction as well as error detection tool during recognition step. This paper has tried to reduce time as well as error rate in proposed work. In this work post- processing of Gurmukhi font has been done in which various methods has been utilized i.e. AVL Tree Implementation, Symmetric delete spelling correction Method, Union of both and Ranking method. A number of techniques are available for detection and correction of errors in recognition of OCR systems, each with its own priorities and weakness. We tried to explore the techniques in order to get good results for recognition rates.*

Keywords: Post Processing, Gurmukhi Font, Optical Character Recognition, AVL Tree, Symmetric delete spelling correction method, Ranking of suggestions

1.Introduction

OCR systems are not able to give correct result in case of handwritten text, because different writing style of the writers. The acknowledgment procedure checking arrangement of characters and supplanted them, if necessary, by a post processor system, for the rightness of the succession and contain important words. One of the issue region for handwritten gurmukhi script is comparative structure of letter sets. Sometime Deviation in acknowledgment results might be on the grounds that the words appear to be comparable however the implications of the words are entirely distinctive appeared in figure 1 Due to expansive variety in writing styles. The acknowledgment procedure may have recognized the word in an unexpected way, whether given word i

ਸਵਾ (SVA) or ਮਵਾ (MVA)

(a)

Figure 1: Both words having comparative structure yet distinctive importance.

To Find Correct acknowledgment of characters which are fundamentally comparable is a troublesome assignment. OCR does not handle acknowledgment itself. There must be some framework ,which can check the rightness of the outcomes to discover mistakes, for acknowledgment process. Furthermore, if acknowledgment isn't right then it ought to have the capacity to rectify or recommend some redress. Utilization of such framework to f enhance general acknowledgment rate for OCR yield.

The goal of this exploration work is to utilized distinctive post preparing strategies for handwritten Gurmukhi Script. The framework determines the vagueness issue to diminishes the acknowledgment recognition rate for gurmukhi script.

Whatever is left of the paper has been sorted out as takes after: segment 2 Review of Literature ,segment 3 covers the proposed research work, results and discourse are incorporated into area segment 4,conclusion.future extension and References finished up in segment 5

2. Review of Literature

Optical Character Recognition is extremely valuable and imperative for different application territories. OCR do work in office robotization, programmed information section in banks, libraries and post workplaces and interactive media plan and so on. These are various languages like Gurmukhi, Devanagari and Bangla for which a completely different OCR is built.

There are different post processing Techniques have been used by researcher. OCR error detection and correction technique have developed by Chaudhuri and Pal for highly inflectional language script like Bangla. Root words and suffixes are two separate lexicons used by the authors, input word are detected by candidate root-suffix pairs, their grammatical agreements and the root/suffix part in which the error has occurred are tested . The fast dictionary access technique is used for the correction, strings which are alternative are generated for an erroneous word. corresponding error of the input string

Levenshtein distance are used for correcting errors, ones Among the alternative strings, those satisfying grammatical agreement in root-suffix and also having smallest. The system has generated output which have an accuracy of 75.61%.
Lehal et. al built up a shape based post processor for Gurmukhi OCR. This depends on the size and state of a word. The most generally common words was consolidated to outline the post processor. There was two apportioned corpus into levels: for first level of corpus incorporate part of 7 disjoint subsets in view of the word length. The second level incorporate state of word was utilized to further section of subsets into segments that were utilized for ID of words which seem to be comparable. Prior to this work incorporate post processing strategy was 94.35% acknowledgment precision for the OCR, which was expanded to 97.34% after the post-processor to the perceived content.

Bansal and Sinha depicted an adjustment technique which utilized a partitioned word lexicon for optically read Devanagari character strings. The separation framework is utilized as a part of search procedure which included assignment. The acknowledgment execution which got a change of roughly 20% in the examination work .This incorporate two methodologies for rightness of the spelling of a word. The another methodology incorporate the accuracy enhancement.

Wing Seong wong proposed a technique which incorporate novel methodology for consolidating OCR innovation inside a structure preparing environment with a specific end goal to separate logical signal words. The Cursive Script Recognition (CSR) framework used to give logical data. The proposed technique has a general change of 12% when contrasted with a structure preparing framework which not used the relevant data.

Hyuk-Chul Kwon has built up a post processor for post processing of a Korean OCR framework utilizing semantic limitations for relevant. It affirms the word by the etymological requirements as collocations of words subsequent to selecting an attainable word,. one, two and three syllable words contain collocation between words is connected in light of the fact that such words are as often as possible not chose. The framework enhanced the word acknowledgment rate of OCR framework 94.72%.

The work created by Shang-Lin Heifer redressing acknowledgment consequences of printed Chinese characters and the point is to amending the recognition, that happened when the most elevated positioned character is not there in a competitor set ,this is in printed Chinese character acknowledgment. The purposed work asserted that 75% of the acknowledgment mistakes for ordinary quality archives and 48% for low quality by utilizing the strategy.

3. Proposed solution

In this work, the post-processing method is used for recognition of Gurmukhi script by reducing errors in the output and to obtain more accuracy in handwritten Gurmukhi OCR.There are four phases included in the research process:
First phase of research work is Shape Encoded Matching include concept of Avl tree implementation and Levenshtein distance for finding relevant suggestions, Second phase include Symmetric delete spelling correction algorithm. Which use the concept of edit distance to find suggestions, after that union is performed in forth phase of research work. which include union of both algorithm Avl tree implementation and Symmetric delete spelling correction algorithm, in last phase of research work where Ranking of suggestions is done, which based on concept of soundex approach to get correct suggestion on first rank
The goal of the post processing is to right errors by spelling redress in the gurmukhi script by utilizing different techniques.

As appeared in Figure 1, There are different writing styles, words which are fundamentally same have diverse significance. In these cases, it is troublesome for OCR to accurately locate the important word.

The proposed technique tackles the vagueness issue up to a specific level by utilizing diverse strategies. this incorporates Gurmukhi characters and lexicon of right words. one of the post handling strategy which contains the plan of partitioning the letters in order of Gurmukhi script into 9 sets in view of their shape comparability. The characters are set in same set taking into account the shape likeness , OCR may remember one character in various ways e.g. ਪ and ਮ, and ਸ have little distinction from each other. The issue turns out to be more unpredictable if there should be an occurrence of manually written content. Following are the steps to perform post processing by different methods:

Phase1:Shape Encoded Matching Algorithm
AVL Tree Implementation
Step 1.1: Database loading & subsets generation

At the starting of proposed work, we upload a text file from database which have around 1, 15,000 Gurmukhi words. We need to define Set with their corresponding codes,. There are 9 set code have been identified for Gurmukhi script as given in table 1. The consonants of Gurmukhi script are divided into 9 Set code for the shape similarity of word. the words are passed through Sets of consonants and replace the words with Set code which matched with them. Below Table 1shows the subset of Consonants defined with its codes from 1 to 9.

Table 1:words with their corresponding codes

Set Code	Set of Consonants
1	ੳ ,ੳ
2	ਚ ,ਟ, ਰ, ਹ ,ਢ ,ਦ, ਫ, ਛ, ਏ ,ਝ, ਵ ਛ
3	ਜ, ਜ਼
4	ਖ ,ਖ਼, ਬ, ਧ, ਥ, ਪ ,ਯ, ਮ, ਸ ,ਸ਼
5	ਕ ,ਙ
6	ਗ, ਗ਼
7	ਅ ,ਘ
8	ਤ, ਡ ,ਭ, ੜ
9	ਠ, ਨ, ਲ, ਲ਼

```
For(I = 0; I < allwords;i++)
      Word = allwords(i);
   For(j=0;j<Word.length;j++)
       If Word(j)== Subsets of Consonants
            Word(j) = Subset Code;
       ifend
     forend
forloopend
```

The codes which replace the words will further move to the Step 2 for the dictionary creation of all that words.

Step 1.2: Initial Dictionaries Creations

In Second step, retrieve the unique words from all code words. There are around 35,000 code words.

Moving forward, these words count as nodes for creating the AVL tree. Each node of Avl Tree has unique code where the different words having same code store in a same node .From these nodes the AVL tree would be generated.

Step 1.3: Creation of AVL Tree

Write all the nodes number in the single line in an increment order.

Find out the centre most node from the all nodes to find which node will be count as the head node.

Count the numbers of left and right hand side of the head node and also get the centre of those nodes.

The centre node of the left side nodes would be the leaf of the head node and the centre node of the right sides nodes would be the right leaf of the head nodes.

This leaf nodes would become the head node and check the left and right hand nodes and the same above mentioned procedure would be followed till all the nodes update into the AVL tree.

Step 1.4: Add words to AVL tree

if the word not found, then this new word add to avl tree by updating avl tree .The new word have some code and then that code count as a node. Each node of Avl Tree has unique code where the different words having same code store in a same node. The new word match with node having same code then this word store on that node. This code have some number, instead of changing into the previous AVL tree get all the codes of AVL tree and add the new word into it. Update it into a single line with ascending order and create a new AVL tree.

Step 1.5: Tested word into AVL tree and get output

- Get the code of tested word.
- Go with the AVL tree. Find out the head node if the tested code have the smaller value than head node then the results is in the left sides. Otherwise it will be in right side.
- With this right and left technique, the matching codes will be found soon and the matching code with maximum accuracy are the outputs of the codes.

Phase 2: Symmetric delete spelling correction (SDSC)

SDSC is the algorithms which segregates the testing word and match the word one by one with the database word in the increasing order based on threshold for that word. If the 85% percent of the words are matched then that would be the output. For the concept of 85%, some of the words are going to be deleted from the database words and some words are going to add.

Step 2.1: Load database and testing words

For the SDSC, update the database and testing word. In the database, there would be 1, 15,000 of Gurmukhi words.

Step 2.2: Split the testing and database words for comparison

```
Test Word[] = split(Testing Word);
 For(int I = 0; i<DataBaseWords.length,i++)
          Int a = 0;
         DataWord[] = split(DataBaseWords[i]);
        For (int j = 0;j<DataWord.length;j++)
             If (DataWord[j]==Test Word[j])
                    A = a+1;
             Ifend
         Forend
        If (A>=85% of TestingWord.length)
            Output = Testing Word;
        Ifend
 Forend
```

Phase 3: Union of both algorithm AVL and SDSC

For the hybridization of both AVL and SDSC algorithms, save the results of both algorithms. Update the result files

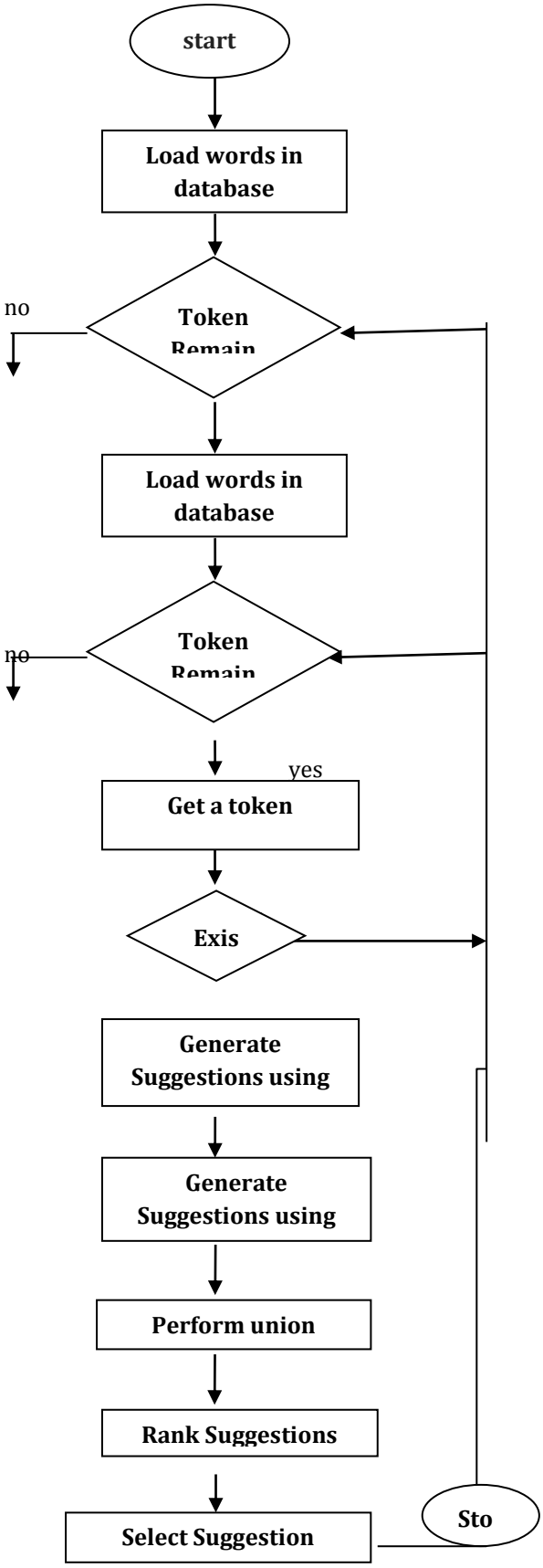

Figure 2: Proposed work Flowchart

```
For (int I = 0;i<AVLResults.length;i++)
 Int a = 0;
   For (int j = 0; j<SDSCResults ; j++)
        If (AVLResults[i]==SDSCResults[j])
       A = 1;
        Ifend
    Forend
      If (A ==1)
        Matched[] = AVLResults[i];
         Else
         Not Matched[] = AVLResults[i];
     Ifend
  Forend
```

Phase 4: Ranking of suggestions

For the ranking of the suggestions which found from the Phase 3, upload the testing word and match word by word to the results the maximum matching suggestion would be count as the upper rank, rest of the suggestion will be lower rank.

3. Results And Discussions

The whole simulation is being done in net beans IDE (8.1) core java environment. Firstly AVL implementation is being done then symmetric delete spelling correction algorithm is applied after that utilization of union method is being done. In the end ranking results has been shown.
Optical character recognition refers to contain written characters. It has been seen that a written document contains lot of spelling mistakes which sometimes completely changes the meaning of the word. The problem of this research work is to perform post processing for OCR generated output written text document to check the spelling of Gurmukhi font. The problem statement also includes a suggestion based on the ranking.

Post processing is a technique to resolve ambiguities problems in OCR generated results by using different methods. These post processing techniques does a lot of improvement for the recognition rate of the Handwritten Gurmukhi Script.

The most commonly used post processing technique is Dictionary look up method . The output is compared to system's built in dictionary which contain various correct words and suggestions are generated. This include According to the difference between the output of the system and the output of the dictionary look up suggestions, the suggestions which are close to the output word are taken as correct and then output sequence of the suitable suggestion are ordered and best suggestion is selected. This is taken into consideration that the suggestions should be valid according to the gurmukhi

grammar rules. These are the following steps follows for the proposed work:

- Uploading of database as text file.
- Match the Test word which is uploaded in another text file with database
- Implementation of AVL tree which include generation of code list for dictionary implementation, which define there is different code for each word of dictionary, Each node of the avl tree have unique code in which less weight function will appear on right hand side and large weight word will appear on right hand side and using Levenshtein distance technique for generating suggestions for misspelled word.
- Symmetric delete spelling correction algorithm is used to generate suggestions. minimum edit distance technique. Match the Test word with the dictionary based on this technique algorithm will be done for spelling checking in which more than 60% threshold value, a word will be contained in file as symmetric file. Now the generated suggestions store in s results
- Combination of generated suggestions by both of Avl tree algorithm and Symmetric delete spelling correction algorithm will be done for spelling checking, performing by union of both result sets.
- Ranking of suggestions will be done using soundex approach in which appropriate suggestion will appear on top of the list of words

INPUT WORD FOR CORRECTION

ਖਰ੍ਹਵਪਣ

Figure 3: Test word

AVL TREE ALGORITHM

ਖਰ੍ਹਵਾਪਣ, ਖਰ੍ਹਵੇਪਣ, ਖੁਰਦਰਾਪਣ, ਖੁਰਦਰੇਪਣ, ਸ਼ਹਿਰਵਾਸੀ,

ਪਰਵਰਿਸ਼, ਸਵੈਵਿਰੋਧੀ, ਪਾਰਦਰਸ਼ੀ, ਬਦਹਵਾਸ, ਬਦਹਵਾਸੀ

Figure 4: AVL Tree results

SYMMETRIC DELETE SPELLING CORRECTION ALGORITHM

ਖਰ੍ਹਵਾਪਣ, ਖਰ੍ਹਵੇਪਣ, ਖਰ੍ਹਵਾਪਣ, ਖਰ੍ਹਵੇਪਣ

Figure 5: Spelling Corrector results

COMMON SUGGESTION IN BOTH ALGORITHM

Common Words in both set Are As Follow:

ਖਰ੍ਹਵਾਪਣ, ਖਰ੍ਹਵਾਪਣ, ਖਰ੍ਹਵੇਪਣ, ਖਰ੍ਹਵੇਪਣ

Non-Common Words are as Follows in set:

ਖੁਰਦਰਾਪਣ,ਖੁਰਦਰੇਪਣ,ਸ਼ਹਿਰਵਾਸੀ

Figure6: Union Results

RANKING OF SUGGESTIONS

Common Words in both set Are As Follow:

ਖਰ੍ਹਵਾਪਣ, ਖਰ੍ਹਵੇਪਣ

Non-Common Words are as Follows in set:

ਖੁਰਦਰਾਪਣ, ਖੁਰਦਰੇਪਣ, ਸ਼ਹਿਰਵਾਸੀ, ਪਰਵਰਿਸ਼,

ਸਵੈਵਿਰੋਧੀ, ਪਾਰਦਰਸ਼ੀ, ਬਦਹਵਾਸ, ਬਦਹਵਾਸੀ

Figure 7: Ranking Results

4. Conclusion And Future Scope

This is a very challenging task of spelling corrector, due to many problems like recognizing of handwriting of different individuals. Every person has its own style of writing, so this is very difficult to recognize characters in each and every style of a writer. In this proposed work this problem will be solved by using three methods one is for spelling checking, second for root generation, third is union of first two methods and other is for ranking algorithm in Gurmukhi script. In other words, the problem of this research work is to perform post processing for any written text document to check the spelling of Gurmukhi word. This work includes a suggestion based on the ranking. It has been seen that proposed method worked well for spelling correction of Gurmukhi text word.

Future scope lies in the usage of various language. The system could perhaps detect real word errors. The automatic word completion could be added as a facility in the system.

References

[1] Munish Kumar, M. K. Jindal, R. K. Sharma, “Classification of Characters and Grading Writers in Offline Handwritten Gurmukhi Script ", accepted for publication in 2011 International Conference on Image Information Processing, IEEE, Vol. 4,pp. 1-4, 2011

[2] Gaurav Singla,Dr.Parmod Kumar," Extract the Punjabi Word with Edge Detector from Machine Printed Document Images", International Journal of Computer Science & Engineering Technology (IJCSET) Vol. 4, pp. 543-545, 2013.

[3] Rajneesh Rani, Renu Dhir, and G.S. Lehal," Identification of Printed Punjabi Words and English Numerals Using Gabor Features.", World Academy of Science, Engineering and Technology, Vol.5, pp. 13-16, 2011.

[4] Anoop Rekha, "Offline Handwritten Gurmukhi Character and Numeral Recognition using Different Feature Sets and Classifiers - A Survey", International Journal of Engineering Research and Applications (IJERA), Vol. 2, No. 3, pp. 187- 191, 2012.

[5] Venkata Reddy, D.Rajeswara Rao, U.Ankaiah, K.Rajesh, "Handwritten Character and Digit Recognition Using Artificial Neural Networks", International Journal of Advanced Research in Computer Science and Software Engineering, Vol.3, No. 4, pp. 140-143, 2013.

[6] Ruby Mehta, Ravneet Kaur, "Neural Network Classifier for Isolated Character Recognition", International Journal of Application or Innovation in Engineering and Management, Vol. 2, No. 1, pp. 285-293, 2013.

[7] Pranob K Charles, V.Harish, M.Swathi, CH. Deepthi, "A Review on the Various Techniques used for Optical Character Recognition", International Journal of Engineering Research and Applications, Vol. 2, No. 1, pp.659-662,, 2012.

[8] Nidhi and V. Gupta, "Punjabi text classification using Naïve Bayes, Centroid and Hybrid Approach", Sundarapandian et al. (Eds): CoNeCo, WiMo, NLP, Vol. 32, pp. 245–252, 2012.

[10] Nidhi and Vishal Gupta. Article: Algorithm for Punjabi Text Classification. International Journal of Computer Applications, Vol. 37, pp.30-35, 2012.

[11] Nidhi and V. Gupta, "Domain based classification of punjabi text documents using ontology and hybrid based approach", Proceedings of the 3rd Workshop on South and Southeast Asian Natural Language Processing (SANLP), COLING, Vol.7, pp. 109–

[12] H. C. Kwon, H. J. Hwang, M. J. Kim, and S. W. Lee, "Contextual post processing of a koreanocr system by linguistic constraints", in proc. Of ICDAR'95, vol. 2, pp557–562, 1995.

[13] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition", Proc. of IEEE, pp 2278–2324, 1998.

.

.